Transactions of the Indian National Academy of Engineering
https://doi.org/10.1007/s41403-020-00090-1

ORIGINAL ARTICLE

Multi-phase MPSP Guidance for Lunar Soft Landing

Avijit Banerjee[1] · **Radhakant Padhi**[1]

Received: 30 December 2019 / Accepted: 23 February 2020
© Indian National Academy of Engineering 2020

Abstract
In this paper, an autonomous optimal guidance algorithm for multi-phase lunar soft landing is presented. Various stringent requirements of a typical multi-phase soft landing are incorporated in guidance formulation. The proposed guidance law is formulated using the model predictive static programming (MPSP), which is an emerging computational guidance algorithm. High accuracy on the terminal position and velocity at the end of each phase is ensured, as the formulation of the MPSP inherently poses terminal output as a set of hard constraints. The spacecraft terminal orientation requirement is embedded in the guidance formulation in a soft constrained manner. Moreover, to facilitate a smooth transition between successive segments, an initial continuity of the guidance command is also attempted in an approximate manner. The effectiveness of the proposed method is demonstrated with simulation results. A processor-in-loop simulation study has been presented to demonstrate the feasibility of the proposed guidance law for possible on-board implementation.

Keywords Multi-phase lunar soft landing · Autonomous guidance · Vertical landing · Optimal control

Introduction

The Moon being the closest celestial body has attracted the global attention for the development of long-term lunar research base (Benaroya and Bernold 2008) for scientific exploration of futuristic objectives. The presence of resources like helium-3 (Taylor and Kulcinski 1999) that are rare on the Earth yet abundant on the Moon makes it preferable in search of future energy resources. Moreover, the recent discovery about the presence of water (Pieters et al. 2009) could add significant support to aid the lunar base a launching platform for interplanetary exploration in the future. However, to accomplish such revolutionary objectives, the autonomous safe landing of a space vehicle is a fundamental prerequisite. Autonomous guidance design for soft landing on a celestial body is quite challenging in general due to complex requirements along the trajectory. A typical autonomous powered soft landing starts from a circular parking orbit of 100 km × 100 km. Once the orbital parameters of this orbit are confirmed from an accurate orbit determination algorithm using the telemetry data and the landing cite is re-confirmed from the on-board imaging, the spacecraft is first guided to a closer elliptical orbit, typically of size 100 km × 18 km, using a Hohmann transfer (Curtis 2013), which is a fuel-efficient maneuver. This elliptic orbit is designed such that the orbital plane contains the designated landing site. Such orbit design confines the motion of spacecraft mostly within the orbital plane during the powered descent maneuver. This strategy avoids additional propellant and control action requirements during this critical phase. However, the guidance should still be developed in 3-D to cater for inaccuracies of the inclination of the initial orbital plane and any cross-plane dispersion during both the Hohmann transfer as well as the powered descent phase. Note that the perilune altitude of 18 km is typically sufficient to avoid collision with mountains present over the lunar surface. Next, at the perilune, spacecraft attains a high orbital velocity. To perform an accurate soft landing, it is necessary to decelerate the spacecraft in a controlled manner and drive it towards the landing site with touchdown velocity. The descent trajectory typically involves three intermediate phases, viz., the initial approach phase, the final approach phase and the terminal descent phase. A schematic diagram of a typical landing trajectory is presented in Fig. 1. Such a multi-phase trajectory has been considered in the recent Chandrayaan-2 mission (Goswami and Annadurai 2011). In addition to the above-mentioned phases, the actual mission

✉ Radhakant Padhi
padhi@iisc.ac.in

[1] Department of Aerospace Engineering, Indian Institute of Science, Bangalore, India

Published online: 18 March 2020

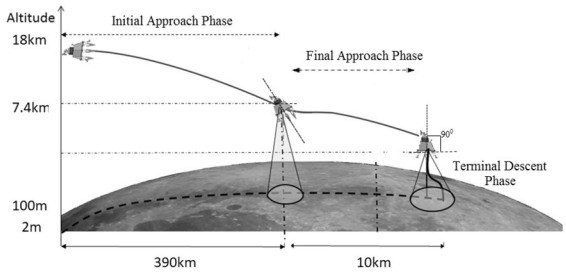

Fig. 1 Schematic representation of a landing scenario

considers an attitude hold phase, which has been operated in a constant-thrust mode without any guidance. Hence, this attitude hold phase is not considered in the present work.

During the initial approach phase, the spacecraft is guided to a relatively low altitude of approximately 7 km and spacecraft velocity decreased significantly. At this point, it is also ensured that the spacecraft orientation is nearly 140° from local horizontal so that accurate navigation information can be extracted from the nadir pointing on-board camera. Next, the final approach phase is initiated, which is a critical phase for the soft landing mission. The objective of the final approach phase is to drive the spacecraft to an altitude of 100 m above the intended landing site and ensure its zero velocity and vertical orientation simultaneously. It can be pointed out that from here onwards direct vertical descent may not be adequate due to the presence of small bolder scattered over the lunar surface. In such a scenario, it is required to autonomously re-target the spacecraft to a relatively safer nearby landing site.

A comprehensive study on the performance assessment of different descent guidance algorithms for planetary soft landing is summarized in Sostaric and Rea (2005). The Apollo missions considered the powered descent guidance with a polynomial approximation of acceleration (Klumpp 1974). The coefficients of acceleration polynomials are evaluated purely based on boundary conditions. Moreover, the guidance law assumes the final time as a tuning parameter. The possible limitations of the Apollo guidance are presented in Lu (2018). An acceleration minimized explicit guidance law (Souza 1997) provides a closed-form acceleration expression. However, this formulation is unable to incorporate the terminal orientation of the spacecraft. Since the orientation constraint appears in the form of terminal acceleration constraint, it is difficult to incorporate in a typical guidance formulation. Introducing the concept of jerk, Uchiyama et al. (2005) formulated an explicit guidance in the two-dimensional framework that is capable of addressing the terminal orientation constraint. However, this guidance formulation considers a simplified spacecraft dynamics, which incorporates various assumptions such as a flat lunar surface, a small variation of downrange angle and constant gravity model. Such assumptions make the applicability of the guidance law restrictive within a small domain of operation.

Considering a complete nonlinear spacecraft equations of motion, various numerical optimal guidance algorithms for lunar soft landing have been presented in the literature. A trajectory optimization-based multi-phase lunar soft landing is presented in Mathavaraj et al. (2017). This formulation accounts for various levels of constraints associated with the individual phases of landing sequences. However, in the presence of perturbations due to sensor inaccuracies and initial condition variation, the trajectory optimization-based offline computed solution may lead to large terminal errors. Hence, the scope of such methods is limited to the initial approach phase only where the terminal accuracy is not a major concern. An iterative guidance law for optimal soft landing which includes parameterization of control vector along with a time scale transformation is proposed in Liu et al. (2008). A generalized MPSP-based guidance design for lunar soft landing is reported in Sachan and Padhi (2019). Adopting an iterative approach, a propellant-optimal powered descent guidance law is presented in Lu (2017). However, this guidance law is unable to incorporate the terminal attitude requirement of spacecraft. Various solution approaches for the fuel-optimal lunar soft landing trajectory design are investigated in Remesh et al. (2016). The simulation results reported in Remesh et al. (2016) demonstrate that the incorporation of spacecraft terminal vertical orientation requirement results in sharp discontinuities (a sudden jump) in spacecraft orientation profiles (e.g., see Fig. 20 of Remesh et al. 2016). Such sudden changes in spacecraft orientation during the flight are not physically realizable by inner-loop autopilot (a detailed autopilot design of the lunar module for a real mission is presented in Rew et al. (2014)). To avoid such sudden changes in orientation requirement, angular rate constraints are incorporated in Ma et al. (2018). The study in Ma et al. (2018) presents a numerical trajectory optimization framework to perform vertical take-off and vertical landing maneuvers of a lunar rover. Adopting an iterative shooting method, a trajectory design for pinpoint soft landing is presented in Cho et al. (2015).

Even though numerical algorithms are powerful in general (as they can handle non-linearity of system dynamics as well as associated constraints), they are usually computationally intensive. It can also be pointed out that 'space-grade processors' have limited computational power. However with the recent advancement of processing capability, numerical guidance algorithms are gaining popularity among researchers (Dueri et al. 2016) and it has enormous scope for next-generation landing missions. In view of that, a special issue (Lu 2017) on computational guidance has been published in the Journal of Guidance, Control and Dynamics. Although the various numerical optimal control methods

are studied for optimal guidance of lunar modules, a few are suitable for on-board implementation due to poor computational efficiency and low convergence rate.

In this context, the MPSP (Halbe et al. 2013) is a finite time numerical nonlinear optimal control algorithm, which is formulated based on the philosophies of the nonlinear model predictive control and approximate dynamic programming. The formulation of MPSP converts the dynamic programming problem into a static optimization problem. Using the symbolic solution of the static costate vector, MPSP results in a closed-form solution of the control update history. The MPSP (Halbe et al. 2013) is a computationally efficient guidance algorithm which iteratively computes the guidance command in real time. Previously, MPSP has been successfully applied for various guidance problems of aerospace applications. The MPSP is gaining popularity due to its computational efficiency and fast convergence property, which makes it a preferable choice as an on-board implementable advanced optimal guidance algorithm. An MPSP-based lunar soft landing is presented in Zhang et al. (2016). However, the formulation presented in Zhang et al. (2016), considers a simplified spacecraft dynamics and it introduces the jerk as control variable, to incorporate the terminal orientation of spacecraft. Moreover, this introduction of additional state variables results in a larger dimension Jacobian matrix, which increases computational burden.

In this paper, an MPSP-based guidance algorithm is proposed for multi-phase autonomous lunar soft landing. It can be mentioned here that the work presented in this paper is an upgraded version of a preliminary study reported in Banerjee et al. (2015). The guidance design presented in Banerjee et al. (2015) adapted a single phase soft landing approach, which ignores the spacecraft terminal orientation requirement. To incorporate various stringent requirements of a typical multi-phase landing trajectory, an MPSP-based guidance algorithm is presented in the current study. The formulation considers the complete nonlinear spacecraft dynamics with spherical lunar model. Terminal position, velocity as well as orientation requirements at end of each segment are explicitly imposed in the guidance formulation to ensure accurate soft landing. Moreover, to facilitate a smooth transition between successive segments, the initial continuity of the guidance command is incorporated. Applicability of the proposed guidance algorithm for possible on-board implementation is verified by processor-in-loop (PIL) simulation. In PIL simulation, the guidance command is evaluated on a target processing environment. In view of computation time requirement, PIL simulation represents a realistic computational environment equivalent to on-board processor. The PIL simulation study reveals a conclusive decision that the proposed MPSP-based soft landing guidance algorithm has enormous hope for the next-generation autonomous soft landing missions.

Problem Formulation for Multi-phase Soft Landing

The translational motion of spacecraft is described using variables primarily defined in two frames, namely the moon-centered inertial frame and a vehicle carried local vertical and local horizontal (LVLH) frame (presented in Fig. 2). The moon-centered inertial frame $X_pY_pZ_p$ is defined with X_p-axis pointing towards the perilune of the elliptical parking orbit, Z_p-axis along the orbit normal and Y_p completes the triad. On the other hand, the local vertical local horizontal (LVLH) frame is located at the center of gravity of the spacecraft. The X_{LVLH}-axis points towards the zenith direction, Z_{LVLH}-axis is directed along the instantaneous orbit normal defined with $\mathbf{r} \times \mathbf{V}_T$ vector and Y_{LVLH}-axis completes the triad. Here, $\mathbf{r}$ denotes the spacecraft position vector (originated from the center of the moon to spacecraft center of gravity) and $\mathbf{V}_T$ represents the instantaneous total velocity vector of the spacecraft. Initially, when descent initiates, the LVLH frame coincides with the inertial frame. As the spacecraft starts maneuvering, X_{LVLH}-axis makes an angle θ with inertial X_P-axis. In general, the elliptic parking orbit is designed in such a way that the orbital plane contains the intended landing site. In an ideal situation (without uncertainties), the entire descent trajectory is aimed to lie within the orbital plane. However, in practice, the spacecraft may perform an out-of-plane maneuver.

Spacecraft Equation of Motion

In the combined inertial and LVLH frames, the equations of motion of the spacecraft (Li et al. 2010) are represented as

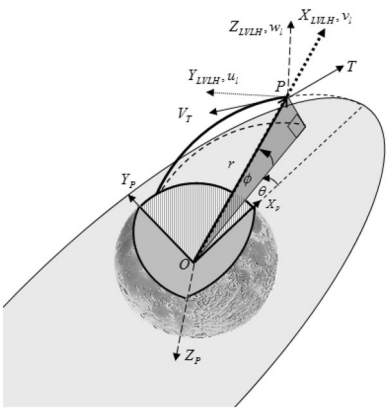

Fig. 2 Axis frames and other variables in a soft landing trajectory

$$\begin{aligned}\dot{r} &= v_l \\ \dot{\theta} &= \frac{u_l}{r\cos\phi} \\ \dot{\phi} &= \frac{w_l}{r} \\ \dot{u_l} &= -\frac{u_l v_l}{r} - \frac{u_l w_l}{r}\tan\phi + \frac{T}{m}\cos\alpha\cos\beta\end{aligned} \tag{1}$$

$$\dot{v_l} = \frac{{u_l}^2}{r} + \frac{{w_l}^2}{r} - \frac{\mu}{r^2} + \frac{T}{m}\cos\alpha\sin\beta \tag{2}$$

$$\begin{aligned}\dot{w_l} &= \frac{{u_l}^2}{r}\tan\phi - \frac{v_l w_l}{r} + \frac{T}{m}\sin\alpha \\ \dot{m} &= -T/(I_{\text{sp}} g_0)\end{aligned} \tag{3}$$

where (r, θ, ϕ) represents the position of spacecraft in the inertial frame of reference; r represents the spacecraft altitude measured from the center of the Moon. The angle θ is used to denote the downrange angle, while the angle ϕ represents the out-of-plane/cross-plane angle. The velocity components u_l, v_l, w_l represent the transverse, radial and cross-plane velocity components of spacecraft along the respective directions of the LVLH frame. The lunar gravitational parameter is represented by μ; m represents the mass of the spacecraft. T denotes the thrust vector magnitude, α and β specify the orientation of thrust vector. I_{sp} is the specific impulse of the propellant and g_0 refers to the Earth's gravitational acceleration at sea level.

Terminal Constraints

Since the spacecraft trajectory is divided into multiple segments (presented in Fig. 1), the guidance formulation needs to ensure various waypoint constraints. At the beginning of each segment, the initial position of the spacecraft is defined as $\left(r_0, \theta_0, \phi_0\right)$. Similarly, the spacecraft initial velocity is denoted as $\left(v_{l0}, u_{l0}, w_{l0}\right)$. To ensure accurate landing sequence, the required terminal positions $\left(r_f, \theta_f, \phi_f\right)$ and the terminal velocity $\left(v_{lf}, u_{lf}, w_{lf}\right)$ need to be ensured at the end of each segment. It can be noted that the spacecraft terminal velocity required for a soft landing is necessarily zero. However, to incorporate the various waypoint constraints for multi-phase guidance design, the initial and terminal velocities are symbiotically presented here.

In addition, the spacecraft's body attitude at the end of various phases needs to satisfy certain terminal orientation constraint to facilitate the camera look angle requirements and to ensure its vertical landing. Since the primary thrust engines are rigidly fixed (strapped down) to the spacecraft, the orientation of spacecraft can be specified in terms of thrust vector orientation α and β. The spacecraft body attitude can be expressed in terms of acceleration components as shown below

$$\alpha = \tan^{-1}\left(\frac{a_w}{\sqrt{a_u^2 + a_v^2}}\right), \quad \beta = \tan^{-1}\left(\frac{a_v}{a_u}\right) \tag{4}$$

where

$$a_u \triangleq \frac{T}{m}\cos\alpha\cos\beta, \quad a_v \triangleq \frac{T}{m}\cos\alpha\sin\beta, \quad a_w \triangleq \frac{T}{m}\sin\alpha \tag{5}$$

Moreover, at the end of various phases, the spacecraft needs to be held against the lunar gravity. This enforces the total terminal acceleration to be constrained by $T_f/m_f = \mu/r^2$. Hence the spacecraft terminal acceleration needs to be enforced to attain a desired terminal value donated as a_{u_f}, a_{v_f} and a_{w_f}. Another important requirement is to facilitate a smooth handover from one phase to other. This necessitates the continuity of acceleration command in between successive stages. Hence, the initial acceleration constraints $a_{u_0}, a_{v_0}, a_{w_0}$ need to be incorporated in the guidance formulation.

Cost Function

To incorporate the initial and terminal acceleration constraints in the guidance formulation, the cost function is selected as

$$J_{\text{T}}^c = J_{\text{IC}} + J_{\text{A}} + J_{\text{TO}} \tag{6}$$

where $\mathbf{a} \triangleq [a_u, a_v, a_w]^{\text{T}}$ represents the acceleration vector and $\mathbf{a}_0, \mathbf{a}_f$ denote the initial and terminal acceleration vectors, respectively. The expression of each components is given by

$$J_{\text{IC}} = \frac{1}{2}\int_0^{t_f}\left(\left(\mathbf{a}-\mathbf{a}_0\right)^{\text{T}} R_1(t)\left(\mathbf{a}-\mathbf{a}_0\right)\right)\text{d}t \tag{7}$$

$$J_{\text{TO}} = \frac{1}{2}\int_0^{t_f}\left(\left(\mathbf{a}-\mathbf{a}_f\right)^{\text{T}} R_3(t)\left(\mathbf{a}-\mathbf{a}_f\right)\right)\text{d}t \tag{8}$$

$$J_{\text{A}} = \frac{1}{2}\int_0^{t_f}\left(\mathbf{a}^T R_2(t)\mathbf{a}\right)\text{d}t \tag{9}$$

The first term J_{IC} is introduced in the cost function to ensure the initial continuity of guidance command in between successive phases. Control weight parameter $R_1(t)$

needs to be tuned such that initially the total cost function J_{T}^{c} is dominated by J_{IC}, as time progress the contribution of $R_1(t)$ decays down to zero. Minimization of J_{IC} essentially enforces the acceleration profile to enforce $\mathbf{a} \rightarrow \mathbf{a}_0$. Similarly, J_{TO} is introduced to incorporate the terminal orientation requirement. The control weight parameter $R_3(t)$ needs to be tuned such that the J_{T}^{a} is dominated by J_{TO} only around the final time and it remains uninfluenced by J_{TO} for the remaining period. As $R_3(t)$ approaches a higher value, $\mathbf{a} \rightarrow \mathbf{a}_f$ at the end of a phase. A representative diagram for the control weight parameter is presented in Fig. 3. The intermediate term J_{A} is introduced to obtain a guidance law that minimizes acceleration requirement. The weighting parameter associated with this term is considered to be a constant for the entire duration.

It can be noted that the acceleration minimization is also a common practice in the available literature (see for e.g., Souza 1997; Guo et al. 2011). Fortunately, however, it turns out that minimization of this cost function does not lead to solutions too far away from the fuel-optimal solution (Zhang et al. 2016). This is because by minimizing the required acceleration components individually also leads to approximate minimization of total acceleration.

To formulate the guidance command that addresses all the above-mentioned requirements in a computationally efficient manner, MPSP guidance has been proposed in this paper. The formulation of MPSP demands the optimal control problem to be described in a discrete-time form. Hence, the soft landing problem has been transformed into an equivalent discrete-time problem. For this purpose, time interval $[0, t_f]$ is divided into a total of N number of equidistant and discrete grid points, where $N = \frac{t_f}{\Delta t} + 1$. Using the Euler discretization method the spacecraft dynamic equations given in Eqs. (1) and (2) are transformed into time discretized form as follows

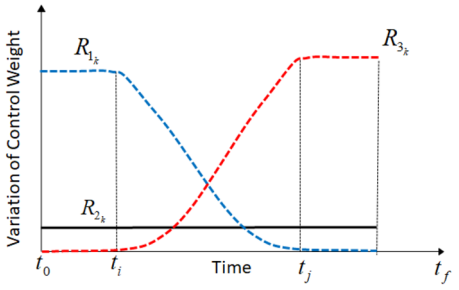

Fig. 3 Schematic representation of control weight parameters

$$\begin{bmatrix} r_{l_{k+1}} \\ \theta_{l_{k+1}} \\ \phi_{l_{k+1}} \\ u_{l_{k+1}} \\ v_{l_{k+1}} \\ w_{l_{k+1}} \\ m_{k+1} \end{bmatrix} = \begin{bmatrix} r_{l_k} + v_{l_k}\Delta t \\ \theta_{l_k} + \left(\frac{u_{l_k}}{r_{l_k}\cos\phi_{l_k}}\right)\Delta t \\ \phi_{l_{k+1}} - \frac{w_{l_k}}{r_{l_k}}\Delta t \\ u_{l_k} + \left(-\frac{u_{l_k}v_{l_k}}{r_{l_k}} - \frac{u_{l_k}w_{l_k}}{r_{l_k}}\tan\phi_{l_k} + a_{u_k}\right)\Delta t \\ v_{l_k} + \left(\frac{{u_{l_k}}^2}{r_{l_k}} + \frac{{w_{l_k}}^2}{r_{l_k}} - \frac{\mu}{{r_{l_k}}^2} + a_{v_k}\right)\Delta t \\ w_{l_k} + \left(\frac{{u_{l_k}}^2}{r_{l_k}}\tan\phi_{l_k} - \frac{v_{l_k}w_{l_k}}{r_{l_k}} + a_{w_k}\right)\Delta t \\ m_k - \frac{m_k\sqrt{{a_{u_k}}^2+{a_{v_k}}^2+{a_{w_k}}^2}}{I_{sp}g_0}\Delta t \end{bmatrix} \tag{10}$$

Here the subscript k is used to denote the k^{th} time instant. To make the problem formulation compatible with MPSP, the cost function (Eq. (6)) is presented in discretized form as follows

$$\begin{aligned} J_{\mathrm{T}}^{d} = &\frac{1}{2}\sum_{k=1}^{N-1}\left(\mathbf{a}_k - \mathbf{a}_0\right)^{\mathrm{T}} R_{1_k}\left(\mathbf{a}_k - \mathbf{a}_0\right) \\ &+ \frac{1}{2}\sum_{k=1}^{N-1}\mathbf{a}_k^{\mathrm{T}} R_{2_k}\mathbf{a}_k^{\mathrm{T}} + \frac{1}{2}\sum_{k=1}^{N-1}\left(\mathbf{a}_k - \mathbf{a}_f\right)^{\mathrm{T}} R_{3_k}\left(\mathbf{a}_k - \mathbf{a}_f\right) \end{aligned} \tag{11}$$

The objective is to obtain the guidance command history $\mathbf{a}_k$ that minimizes the objective function given in Eq. (11), subjected to the system dynamics (presented in Eq. (10)) and terminal constraints imposed due to various waypoint constraints at the end of each phase. A brief description of the MPSP formulation in a generic form is presented next.

Brief Overview of MPSP

The formulation of MPSP considers a generic nonlinear system dynamics in the discrete-time form given as

$$X_{k+1} = F(X_k, U_k), X(t_0) = X_0 \tag{12}$$

$$Y_k = h(X_k) \tag{13}$$

where $X_k \in \Re^n$, $U_k \in \Re^m$, $Y_k \in \Re^p$ represent the system state, input, and output vectors, respectively. X_0 denotes the initial condition. The index k is used to denote the discrete-time instants, which varies from $k = 1, 2, \ldots, N-1$. Starting with a guess time history of the control input U_k^i, the objective is to determine control update $U_k^{i+1} \triangleq U_k^i - dU_k^i$ such that the terminal output Y_N approaches its desired value Y_N^d, while minimizing a cost function

$$J = \sum_{k=1}^{N-1} L\left(U_k\right)$$

Here, dU_k^i represents the update in control input, and U_k^{i+1} denotes the updated control. The super script i is used to indicate the current iteration. The terminal output error is defined as

$$\Delta Y_N^i \triangleq Y_N^i - Y_N^d \tag{14}$$

Since the output is considered as a function of state vector, a Taylor series approximation of Y_N about Y_N^d (limited upto the first order term) yields

$$\Delta Y_N^i = Y_N^i - Y_N^d = \frac{\partial Y_N^i}{\partial X_N^i} dX_N^i \tag{15}$$

where the expression of dX_{k+1}^i is presented as follows

$$dX_{k+1}^i = \left[\frac{\partial F}{\partial X_k^i}\right] dX_k^i + \left[\frac{\partial F}{\partial U_k^i}\right] dU_k^i \tag{16}$$

Substituting the expression of dX_k^i for $k = N-1, N-2, \ldots, 1$ into (16) with necessary algebraic simplification yields

$$\Delta Y_N^i \simeq dY_N^i = B_1 dU_1^i + \cdots + B_{N-1}^i dU_{N-1}^i = \sum_{k=1}^{N-1} B_k^i dU_k^i \tag{17}$$

where

$$B_k^i = \left[\frac{\partial Y_N^i}{\partial X_N^i}\right]\left[\frac{\partial F}{\partial X_{N-1}^i}\right] \cdots \left[\frac{\partial F}{\partial U_k^i}\right] \tag{18}$$

Hence, the terminal output error dY_N is represented as a function of control updates dU_k^i for $k = 1, \ldots, (N-1)$. Moreover, Eq. (17) is under-constrained (since, $(N-1)m >> p$) and hence, there is a scope of optimization. For the purpose of completeness, a generic optimization problem is formulated as follows

$$\min_{U_k^{i+1}} J = \sum_{k=1}^{N-1} L\left(U_k\right) \tag{19}$$

$$\text{subject to } dY_N^i = \sum_{k=1}^{N-1} B_k^i dU_k^i \tag{20}$$

It can be noted here, MPSP transforms the optimal control formulation into a static optimization problem. The optimization problem consists of a set of equality constraints as presented in Eq. (20). The equality constraints essentially ensure the terminal output requirements, which are imposed as a set of hard constraints in the optimization process. The optimization problem with equality constraints can be solved using a constant Lagrange multiplier λ, which represents an equivalent form of costate vector (Bryson 1975). In general, costate vectors are dynamic. However, the MPSP formulation enables static costate-equivalent variables. Moreover, a choice of cost function $L = \left(U_k^{i+1T} R U_k^{i+1}\right)$ results in closed-form control update U_k^{i+1} (More details can be found in Halbe et al. (2013)) The closed-form control update makes the MPSP algorithm computationally efficient. In the present work, the MPSP algorithm has been applied for autonomous guidance design of lunar soft landing. A detail of guidance design for multi-phase lunar soft landing is presented next.

MPSP Guidance for Multi-phase Soft Landing

In the context of lunar soft landing, the system state, output and input vectors (in time discretized form) are considered as follows

$$X_k = \left[r_k \;\; \theta_k \;\; \phi_k \;\; u_{l_k} \;\; v_{l_k} \;\; w_{l_k} \;\; m_k\right]^{\mathrm{T}} \tag{21}$$

$$Y_k = \left[r_k \;\; \theta_k \;\; \phi_k \;\; u_{l_k} \;\; v_{l_k} \;\; w_{l_k}\right] \tag{22}$$

$$U_k = \left[a_{u_k} \;\; a_{v_k} \;\; a_{w_k}\right]^{\mathrm{T}} \tag{23}$$

The objective of the multi-phase guidance design is to obtain the optimal guidance command U_k^* that ensures the terminal output $Y_N \rightarrow Y_N^d$ while minimizing the objective function

$$\begin{aligned} J_{\mathrm{T}} = & \frac{1}{2}\sum_{k=1}^{N-1}\left(U_k^{i+1} - U_0\right)^{\mathrm{T}} R_{1_k}\left(U_k^{i+1} - U_0\right) \\ & + \frac{1}{2}\sum_{k=1}^{N-1}\left(U_k^{i} - dU_k^{i}\right)^{\mathrm{T}} R_{2_k}\left(U_k^{i} - dU_k^{i}\right)^{\mathrm{T}} \\ & + \frac{1}{2}\sum_{k=1}^{N-1}\left(U_k^{i+1} - U_f\right)^{\mathrm{T}} R_{3_k}\left(U_k^{i+1} - U_f\right) \end{aligned} \tag{24}$$

where U_0 and U_f indicate the required initial and terminal acceleration constraints, respectively. The required terminal position and velocity of spacecraft at the end of each segments is denoted by $Y_N^d = \left[r_f \;\; \theta_f \;\; \phi_f \;\; u_{l_f} \;\; v_{l_f} \;\; w_{l_f}\right]$. The MPSP guidance requires an initial guess control history to begin with. Here, an inverse polynomial-based guidance law presented in Banerjee and Padhi (2015) is considered to obtain the initial guess U_k^0. To obtain the updated guidance command U_k^{i+1}, the following optimization problem is formulated

$$\min_{U_k^{i+1}} J_{\mathrm{T}} \tag{25}$$

$$\text{subject to } dY_N^i = \sum_{k=1}^{N-1} B_k^i dU_k^i \tag{26}$$

It can be noted that one can symbolically obtain the expressions of Jacobian matrices $\left[\frac{\partial F}{\partial X_k}\right]$ and $\left[\frac{\partial F}{\partial U_k^i}\right]$, which are required to compute the sensitivity matrices B_k^i involved in MPSP formulation. The above optimization problem with equality constraint can be solved by minimizing the following free optimization problem with the augmented cost function given as

$$\bar{J}_{\mathrm{T}} = J_{\mathrm{T}} + {\lambda^{i+1}}^{\mathrm{T}}\left(\sum_{k=1}^{N-1} B_k^i dU_k^i - dY_N^i\right) \tag{27}$$

where λ^{i+1} represents a static costate variable. The following equations are obtained from the necessary conditions for optimality, given as

$$\begin{aligned}\frac{\partial \bar{J}_{\mathrm{T}}}{\partial U_k^{i+1}} &= R_{1_k}\left(U_k^{i+1} - U_0\right) + R_{2_k} U_k^{i+1} \\ &\quad + R_{3_k}\left(U_k^{i+1} - U_f\right) + B_k^{iT}\lambda^{i+1} = 0\end{aligned} \tag{28}$$

$$\frac{\partial \bar{J}_{\mathrm{T}}}{\partial \lambda^{i+1}} = \sum_{k=1}^{N-1} B_k^i\left(U_k^i - U_k^{i+1}\right) - dY_N^i = 0 \tag{29}$$

Rearranging Eq. (28), the expression U_k^{i+1} is obtained in terms of λ^{i+1}, given as

$$U_k^{i+1} = -{R_{s_k}}^{-1} B_k^{iT}\lambda^{i+1} + {R_{s_k}}^{-1}\left(R_{3_k}U_f + R_{1_k}U_0\right) \tag{30}$$

where,

$$R_{s_k} = \left(R_{1_k} + R_{2_k} + R_{3_k}\right)$$

Substituting the expression of U_k^{i+1} into Eq. (29), an explicit expression of λ^{i+1} is obtained as

$$\lambda^{i+1} = A_{\lambda_m}^{i^{-1}}\left(dY_N^i - B_{\lambda_m}^i\right) \tag{31}$$

where

$$A_{\lambda_m}^i = \left(\sum_{k=1}^{N-1} B_k^i {R_{s_k}}^{-1} B_k^{iT}\right) \tag{32}$$

$$B_{\lambda_m}^i = \left(\sum_{k=1}^{N-1} B_k^i {R_{s_k}}^{-1}\left(R_{3_k}U_f + R_{1_k}U_0\right) - \sum_{k=1}^{N-1} B_k^i U_k^i\right) \tag{33}$$

Substituting expression of λ^{i+1} from Eq. (31) into Eq. (30), the closed-form expression of the control update is obtained as

$$\begin{aligned}U_k^{i+1} &= -{R_{s_k}}^{-1} B_k^{iT} A_{\lambda_m}^{i^{-1}}\left(dY_N^i - B_{\lambda_m}^i\right) \\ &\quad + {R_{s_k}}^{-1}\left(R_{3_k}U_f + R_{1_k}U_0\right)\end{aligned} \tag{34}$$

It can be noted here, the expression of the closed-form control update presented in Eq. (34) is different from Halbe et al. (2013). This is due to the fact that the formulation presented here considers a cost function which is different from the classical formulation (Halbe et al. 2013). The modification in the cost function is incorporated to address the initial and terminal acceleration constraints, which are necessary for accurate soft landing. The guidance command computation steps are summarized as follows

Guidance Command Computation Steps

1. The initial guess for guidance computation (i.e. $U_k^i = \begin{bmatrix} a_{u_k}^i & a_{v_k}^i & a_{w_k}^i \end{bmatrix}^{\mathrm{T}}$ for $k = 1, 2, \ldots, N-1$) is generated using an explicit guidance law presented in Banerjee and Padhi (2015).
2. Based on the current initial condition X_0 and the initial guess control history, the point mass spacecraft dynamics presented in Eqs. (1) and (2) is propagated in forward direction. The propagation uses numerical integration method. The terminal output error dY_N^i is evaluated as follows

$$dY_N^i = \begin{bmatrix} r_N^i - r_f & \theta_N^i - \theta_f & \phi_N^i - phi_f & u_{l_N}^i - u_{l_f} & v_{l_N}^i - v_{l_f} & w_{l_N}^i - w_{l_f} \end{bmatrix}^{\mathrm{T}} \tag{35}$$

3. If $|dY_N^i| > \epsilon$ (a small tolerance value), the guidance command $a_{u_k}^i$, $a_{v_k}^i$ and $a_{w_k}^i$ needs to be corrected. To compute the updated acceleration, recursively, evaluate the sensitivity matrix B_k^i at each grid points. The computation of B_k requires Jacobian matrices $\left[\frac{\partial F_k}{\partial X_k}\right]_{7\times7}$, $\left[\frac{\partial F_k}{\partial U_k}\right]_{7\times3}$, $\left[\frac{\partial Y_N}{\partial X_N}\right]_{6\times7}$, which are obtained symbolically.
4. The updated guidance command is computed from Eq. (34).
5. The updated control history is used to propagate the system dynamics presented in Eqs. (1) and (2). Based on the propagation the norm of terminal output error $|dY_N|$ is evaluated. The above steps are continued until the terminal output error is obtained within a small tolerance bound.

Simulation Results

To demonstrate the effectiveness of the MPSP-based guidance design, a typical multi-phase soft landing scenario is considered in the simulation study. The landing sequence

Table 1 Initial and terminal conditions for the initial approach phase

Variables	Nominal initial conditions	Variation of initial conditions	Terminal conditions
Altitude (km)	18	± 2	7.5
Downrange (km)	0 m	± 0.5	372
Cross-range (m)	0	± 200	0
Transverse velocity (m/s)	1690	± 5	200
Radial velocity (m/s)	0	± 5	− 35
Out-of-plane velocity (m/s)	0	± 2	0

incorporates the initial approach phase, the final approach phase and the terminal descent phase. The nominal boundary conditions for the various phases are mentioned in Tables 1 and 2 respectively. For simulation study, the Moon radius is considered as 1737 km, lunar gravitational parameter $\mu = 4.9048 \times 10^{12} \mathrm{m}^3\mathrm{s}^{-3}$, and engine-specific impulse $I_{\mathrm{sp}} = 312s$

Guidance Design for the Initial Approach Phase

Since the powered descent commence with the initial approach phase, the requirement of continuity of guidance command is not applicable here. Hence, the cost function considered for the initial approach phase ignores the term J_{IC} by imposing $R_{1_k} = 0_{3\times3}$ matrix. However, the terminal orientation of the spacecraft is incorporated with appropriate weighting matrices associated with J_{TO}. It is to be noted that the camera look angle requirement demands the terminal orientation to be $\alpha_f = 0^0, \beta_f = 140^0$. The MPSP formulation with terminal acceleration constraint demands the terminal acceleration to be specified. To obtain equivalent terminal acceleration $\frac{T_f}{m_f}$, the following design consideration is adopted.

An additional condition is imposed as $\dot{u}_{l_f} = 0$, which also enables the transverse velocity profile to smoothly reach its desired terminal value. Substituting $\dot{u}_{l_f} = 0$ at $t = t_f$ into Eq. (2), the required terminal acceleration is obtained as

$$\frac{T_f}{m_f} = \frac{1}{(\cos\alpha_f \cos\beta_f)}\left(\frac{u_{l_f} v_{l_f}}{r_f} + \frac{u_{l_f} w_{l_f}}{r_f}\tan\phi_f\right) \tag{36}$$

Using the computed $\frac{T_f}{m_f}$ along with the required α_f, β_f the terminal acceleration U_0 is imposed. Since the initial spacecraft velocity is relatively high, starting from the perilune it is difficult to target towards a desired terminal orientation of the spacecraft. Moreover, such practice ends up with high propellant consumption. During the powered descent, as the spacecraft velocity significantly decreases, it is relatively easy to target for a desired spacecraft orientation. In view of the above analysis, the diagonal elements of control weight

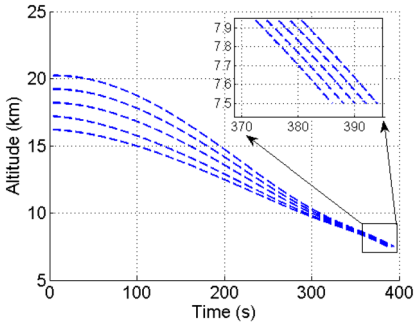

Fig. 4 Altitude profiles for the initial approach phase

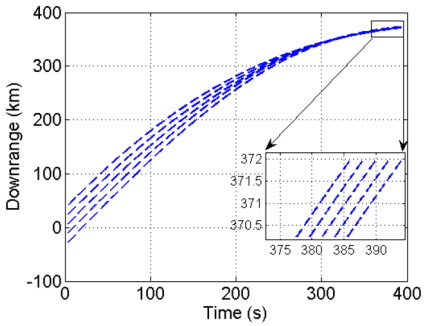

Fig. 5 Downrange profiles for the initial approach phase

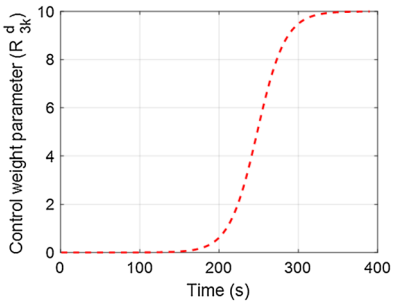

Fig. 6 Variation of control weight parameter $R^d_{3_k}$ for the initial approach phase

parameter R_{3_k} are designed such that initially it remains zero, as time elapsed it starts building up and becomes high at the end of this phase as shown in Fig. 6. All the three diagonal elements of R_{3_k} is considered to have equal weighting. The control weighting matrix R_{2_k} is chosen to be identity $R_{2_k} = I_{3\times3}$. The variation of the spacecraft altitude profiles is shown in Fig. 4. It is evident that the spacecraft gradually descend from 18km altitude to 7.5 km and also it covers the required downrange as shown in Fig. 5. The spacecraft velocity profiles are presented in Figs. 7 and 8. The required terminal velocities are achieved. The spacecraft out-of-plane

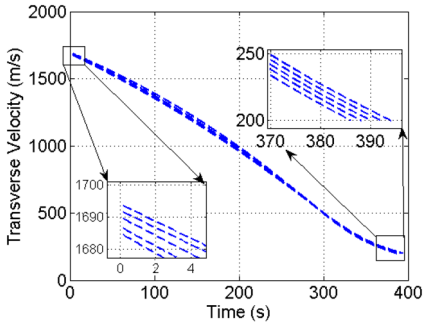

Fig. 7 Transverse velocity profiles for the initial approach phase

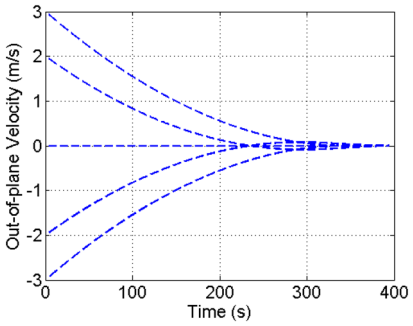

Fig. 10 Variation of out-of-plane velocity for the initial approach phase

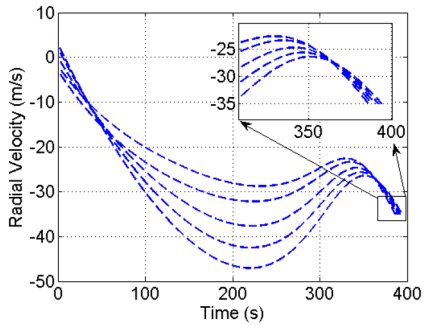

Fig. 8 Radial velocity profiles for the initial approach phase

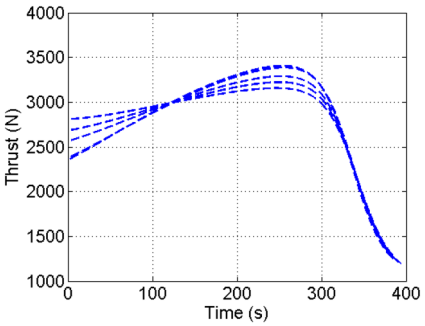

Fig. 11 Variation of thrust magnitude for the initial approach phase

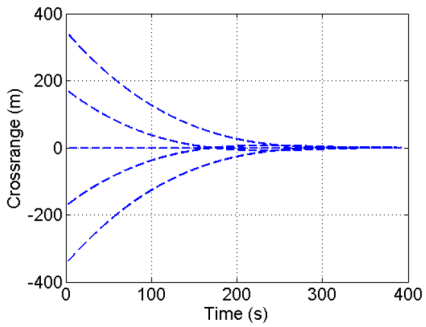

Fig. 9 Variation of cross-range for the initial approach phase

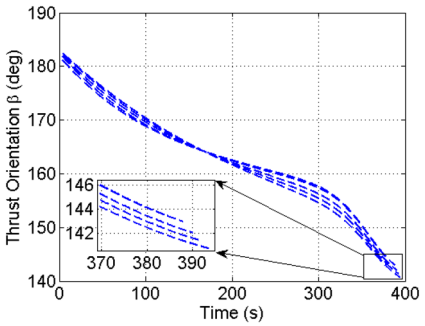

Fig. 12 Variation of β for the initial approach phase

maneuver is presented in Figs. 9 and 10. It is evident that the guidance law is capable of bringing the spacecraft back into the orbital plane and ensure that terminal cross-range and out-of-plane velocity are nullified.

The time variation of the thrust vector magnitude and its orientation are presented in Figs. 11 and 12. The terminal spacecraft orientation is approximately achieved within an error margin of 3^0 on β_f and $\pm 0.5^0$ on α_f. The simulation study for the final approach phase is presented next.

Guidance Design for the Final Approach and Terminal Descent Phase

The guidance design for both the final approach phase and the terminal descent phase needs to ensure the initial continuity of guidance command as well as the terminal orientation requirement. Hence, the guidance law for these phases considers the cost function as presented in Eq. (24), where the enforcement on both the initial and terminal acceleration is incorporated. The boundary conditions for these phases are mentioned in Table 2. Here, the boundary conditions are

Table 2 Initial and terminal conditions for the final approach and terminal descent phases

Variables	Final approach phase		Terminal descent phase
	Initial conditions	Terminal conditions	Terminal conditions
Altitude (m)	6000	100	2
Downrange (m)	− 10,000	0	60
Crossrange (m)	0	0	0
Transverse velocity	144	0	0
Radial velocity	− 44	0	0
out-of-plane velocity	0	0	0
Thrust orientation β	140°	90°	90°
Thrust orientation α	0°	0°	0°

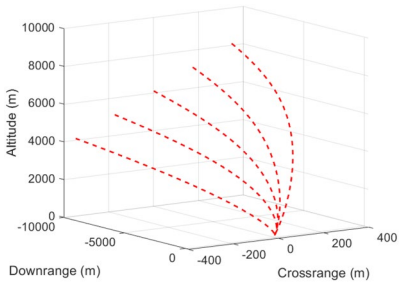

Fig. 13 Variation of spacecraft trajectory

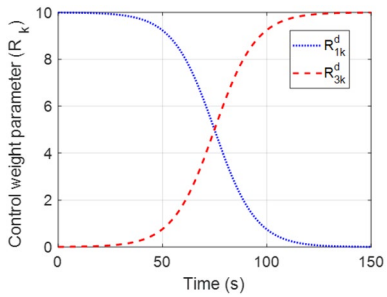

Fig. 14 Variation of control weight parameters for final approach phase

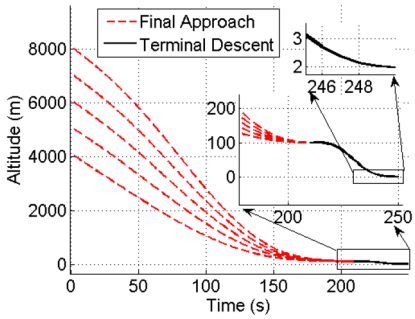

Fig. 15 Altitude profile for the final approach and terminal descent phase

specified with respect to the local landing site-based frame of reference for meaningful visualization (Fig. 13).

The weighting parameters R_{1_k} and R_{3_k} are designed to ensure the required initial continuity and terminal orientation of spacecraft. The shape of $R^d_{1_k}$ and $R^d_{3_k}$ is presented in Fig. 14. It can be noted here the design parameter $R^d_{1_k}$ is initially high compared to $R^d_{3_k}$. It has been designed in such a manner as to ensure that the total cost function J_T is dominated by J_{IC}. As time progress, the magnitude of $R^d_{1_k}$ gradually decreases, and hence the effect of J_{IC} gradually diminishes. Such a design of $R^d_{1_k}$ initially enforces the optimization problem to minimize the difference between U_k and U_0 and thereby, it ensures the initial continuity of guidance command. Here, the required U_0 is computed from the terminal acceleration of the previous segment. $R^d_{1_k}$ and $R^d_{3_k}$ essentially represent the diagonal elements of the corresponding matrices. All the three diagonal elements of individual weighting matrices are considered to have equal weighting. The $R^d_{2_k}$ is set as unity throughout time.

Accurate soft landing demands $\beta_f = 90^0, \alpha_f = 0^0$ to ensure the vertical orientation of the spacecraft, and $\frac{T_f}{m_f} = g$ is necessary to hold the vehicle against lunar gravity. The weighting matrix $R^d_{3_k}$ is designed such that initially, the static optimization is ignorant of the term J_{TO}. As time progress, the magnitude of $R^d_{3_k}$ gradually increases and the effect of J_{TO} is realized. Such design of $R^d_{3_k}$ minimizes the difference between U_k and U_f to ensure $U_k \rightarrow U_f$ at final time.

The variation of spacecraft altitude profiles is shown in Fig. 15. One can observe that the spacecraft altitude smoothly approaches 100 m at the end of final approach phase and eventually at the end of the terminal descent phase, spacecraft altitude reaches the desired hight of 2 m. The variation of downrange profiles is shown in Fig. 16. The remaining downrange of about 10 km is covered in

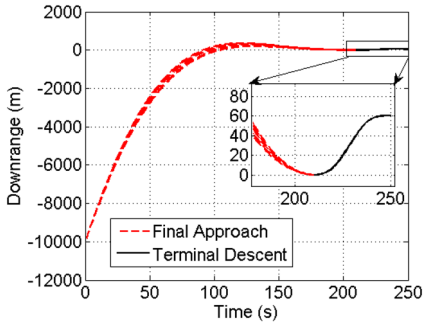

Fig. 16 Downrange profile for the final approach and terminal descent phase

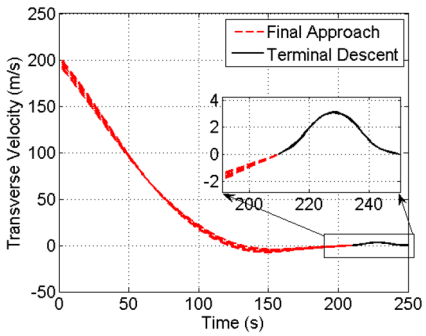

Fig. 17 Transverse velocity profile for the final approach and terminal descent phase

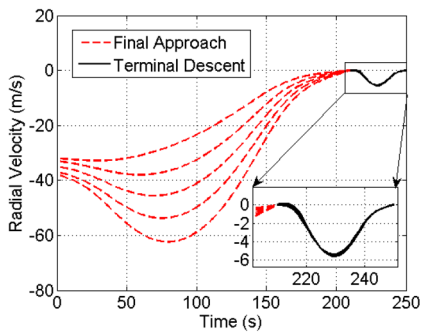

Fig. 18 Radial velocity profile for the final approach and terminal descent phase

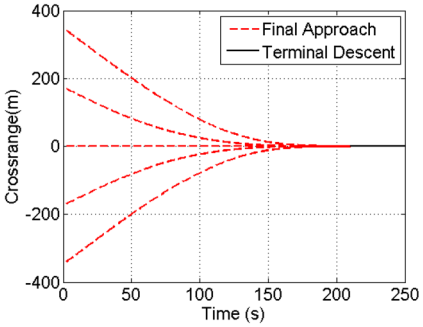

Fig. 19 Variation of cross-range for the final approach and terminal descent phase

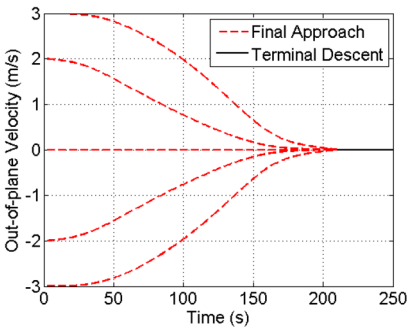

Fig. 20 Variation of out-of-plane velocity for the final approach and terminal descent phase

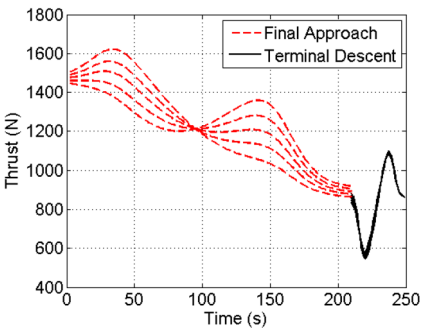

Fig. 21 Variation of thrust magnitude for the final approach and terminal descent phase

the final approach phase. To demonstrate the capability of re-targeting maneuver, a terminal downrange requirement of 60 m has been considered for the terminal descent phase. The velocity profiles are presented in Figs. 17 and 18. It can be observed that the spacecraft terminal zero velocity is achieved. A possible out-of-plane variation is presented in Figs. 19 and 20. The guidance formulation in $3-D$ framework is capable of encountering the spacecraft out-of-plane motion and enforces the spacecraft to cover the required cross-range. The variation of thrust vector magnitude and its orientation are shown in Figs. 21, 22, 23. The initial orientation of the spacecraft is enforced to be around $\beta_0 = 140^0$. The required terminal orientation is $U_k \rightarrow U_f$ at the end of a phase achieved within an error bound of $\pm 2^0$. Spacecraft trajectories targeted to different landing sites are shown in Fig. 24, which also demonstrates the on-board re-targeting capability of the proposed guidance law. Typically, guidance computation requires 3–4 iteration for its convergence and the convergence time in a desktop environment is reasonably fast. To test

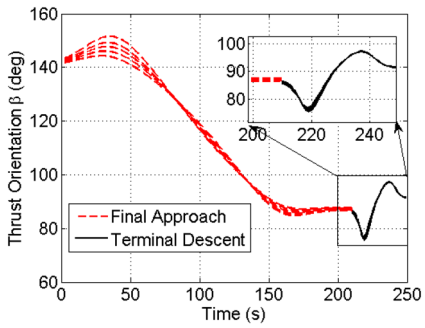

Fig. 22 Variation of β profile for the final approach and terminal descent phase

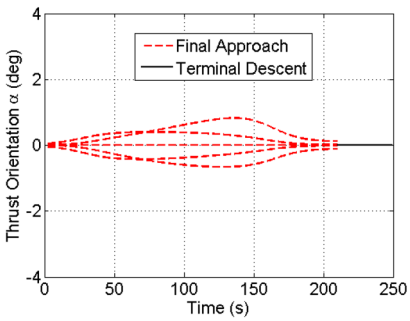

Fig. 23 Variation of α profile for the final approach and terminal descent phase

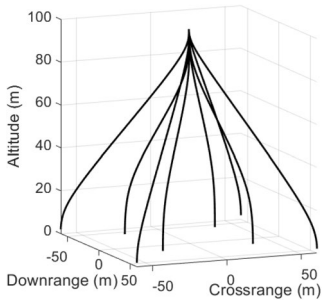

Fig. 24 Spacecraft re-targeting maneuver

the computational performance of MPSP guidance on a space-grade processor, a Processor-in-Loop (PIL) simulation study has been carried out, which is presented next.

Processor-in-Loop Simulation

The proposed MPSP guidance is iterative in nature. To validate the possibility of the proposed guidance algorithm for on-board implementation, the PIL simulation has been carried out in a non real-time environment. The initial and final approach phases of lunar soft landing have been considered here for the demonstration. The computational efficiency

Fig. 25 PIL simulation setup in Aerospace Simulation Lab, IISc

of guidance algorithm has been evaluated in terms of computational time, required to evaluate a guidance command. The Texas Instruments (TI) *TMS*320*C*6748 digital signal processor (DSP) has been used in this experiment. To access the internal memory of the processor, a targeted interface called 'Code Composer Studio (CCS) Integrated development environment' (Texas Instruments 2015) is used. The processor is mounted on a circuit board, called Low-Cost Development Kit (LCDK) (Reay 2012). A regulated 5 V, 2A power supply is used to power the board. An external emulator named XDS100v2 JTAG is used as connecting probe to communicate with the CCS from a host computer through the USB interface. The laboratory experimental setup is shown in Fig. 25.

To carry out the PIL simulation, the proposed soft landing guidance algorithm has been coded using 'C' programming language. The 'C' code is placed at an internal memory, which is of size 256 KB. The guidance command is computed using the dedicated DSP processor. The processor supports a clock rate of 450 Mhz. The total number of CPU cycle required for the computation of each iterations is recorded. The computation time in the PIL environment is estimated from the following relation presented as

$$\text{Exicution time (ms)} = \frac{\text{Number of CPU cycles}}{\text{Processor clock frequency}} \tag{37}$$

The evaluation of computation time in a real-time processing environment is presented in Tables 3 and 4. The results are presented in terms of CPU cycles counts required to complete one iteration and CPU cycle taken to complete its convergence. The CPU cycles per iteration results are

Table 3 PIL simulation for the initial approach phase

No. of iterations	CPU cycles	450 MHz (ms)	1.0 GHz (ms)
1	32,482,218	72.18	32.48
3	97,446,654	216.54	97.44

Table 4 PIL simulation: initial final phase

No. of Iterations	CPU cycles	450 MHz (ms)	1.0 GHz (ms)
1	24,357,110	54.12	24.35
3	73,071,152	162.38	73.04

significant. Moreover, considering the fact that the count of CPU cycles for an algorithm is invariant over processing environment, approximate computation time on a 1 GHz processing environment is evaluated.

Typically, closed-loop guidance operates on 100 ms guidance cycle. Hence, all the necessary computations to evaluate a guidance command needs to be completed within 100 ms. Note that, the 'space-grade' processors that are presently being used have a clock frequency of the order of 450 MHz. However, the computational efficiency of next-generation space-grade processors (Lovelly and George 2017) are also getting improved with the remarkable advancement of modern research in the field of electronic devices. Hence, next-generation space-grade processors are expected to support up to 2.0 GHz clock frequencies. Furthermore, it is evident from the results presented in Tables 3 and 4 that using 1 GHz processor the algorithm takes less than 100 ms (for both initial as well as final approach phases). In view of that, the proposed MPSP-based soft landing guidance has enormous hope for the next-generation soft landing missions.

Conclusion

In this paper, MPSP guidance algorithm for multi-phase autonomous lunar soft landing is proposed. Various stringent requirements of typical multi-phase soft landing are addressed. High accuracy on the terminal position and velocity is ensured at the end of each phase. Moreover, the initial control continuity as well as terminal orientation requirements are accounted in the guidance formulation. The effectiveness of the proposed guidance algorithm is demonstrated using simulation results.

Applicability of the proposed guidance algorithm for possible on-board implementation is verified by processor-in-loop (PIL) simulation. The PIL simulation study reveals a conclusive decision that the proposed MPSP-based soft landing guidance algorithm has enormous hope for the next-generation autonomous soft landing missions. Moreover, the formulation considers a generic nonlinear spacecraft dynamics and hence it can address a similar class of autonomous soft landing problems.

Funding This article was funded by Indian Space Research Organisation (Grant no. CSSP/ISTC0346).

References

Banerjee A, Padhi R (2015) Inverse polynomial based explicit guidance for lunar soft landing during powered braking. In: 2015 IEEE conference on control applications (CCA) (IEEE), pp 768–773

Banerjee A, Padhi R, Vatsal V (2015) Optimal guidance for accurate lunar soft landing with minimum fuel consumption using model predictive static programming. In: 2015 American control conference (ACC) (IEEE), pp 1861–1866

Benaroya H, Bernold L (2008) Engineering of lunar bases. Acta Astronaut 62(4–5):277

Bryson AE (1975) Applied optimal control: optimization, estimation and control. CRC, Boca Raton

Cho DH, Kim D, Leeghim H (2015) Optimal lunar landing trajectory design for hybrid engine. In: Mathematical problems in engineering, pp 1–8

Curtis HD (2013) Orbital mechanics for engineering students. Butterworth-Heinemann, Oxford

Dueri D, Açıkmeşe B, Scharf DP, Harris MW (2016) Customized real-time interior-point methods for onboard powered-descent guidance. J Guid Control Dyn 40(2):197

Goswami JN, Annadurai M (2011) Chandrayaan-2 mission. In: Lunar and planetary science conference, vol 42, p 2042

Guo Y, Hawkins M, Wie B (2011) Optimal feedback guidance algorithms for planetary landing and asteroid intercept. In: Proceedings of AAS/AIAA astrodynamics specialist conference, pp 569–588

Halbe O, Raja RG, Padhi R (2013) Robust reentry guidance of a reusable launch vehicle using model predictive static programming. J Guid Control Dyn 37(1):134

Klumpp AR (1974) Apollo lunar descent guidance. Automatica 10(2):133

Li M, Macdonald M, McInnes C, Jing W (2010) Analytical landing trajectories for embedded autonomy. Proc Inst Mech Eng Part G J Aerosp Eng 224(11):1177

Liu XL, Duan GR, Teo KL (2008) Optimal soft landing control for moon lander. Automatica 44(4):1097

Lovelly TM, George AD (2017) Comparative analysis of present and future space-grade processors with device metrics. J Aerosp Inf Syst 14(3):184

Lu P (2017) Introducing computational guidance and control

Lu P (2017) Propellant-optimal powered descent guidance. J Guid Control Dyn 41(4):813

Lu P (2018) Augmented apollo powered descent guidance. J Guid Control Dyn 42(3):447

Ma L, Wang K, Xu Z, Shao Z, Song Z, Biegler LT (2018) Trajectory optimization for lunar rover performing vertical takeoff vertical landing maneuvers in the presence of terrain. Acta Astronaut 146:289

Mathavaraj S, Pandiyan R, Padhi R (2017) Constrained optimal multi-phase lunar landing trajectory with minimum fuel consumption. Adv Sp Res 60(11):2477

Pieters CM, Goswami J, Clark R, Annadurai M, Boardman J, Buratti B, Combe JP, Dyar M, Green R, Head J et al (2009) Character and spatial distribution of OH/H2O on the surface of the Moon seen by M3 on Chandrayaan-1. Science 326(5952):568

Reay DS (2012) OMAP-L138 low cost development kit (LCDK) for hands-on DSP teaching. In: 2012 5th European DSP education and research conference (EDERC) (IEEE), pp 164–168

Remesh N, Ramanan R, Lalithambika V (2016) Fuel optimum lunar soft landing trajectory design using different solution schemes. Int IREASE 9(5):131

Rew DY, Ju G, Lee S, Kim K, Kang SW, Lee SR (2014) Control system design of the Korean lunar lander demonstrator. Acta Astronaut 94(1):328

Sachan K, Padhi R (2019) Waypoint Constrained Multi-Phase Optimal Guidance of Spacecraft for Soft Lunar Landing. Unmanned Syst 7(02):83

Sostaric R, Rea J (2005) Powered descent guidance methods for the moon and mars. In: Proceedings of AIAA guidance, navigation, and control conference and exhibit (San Francisco, California, 2005), AIAA Paper 2005-6287

Souza CD (1997) An optimal guidance law for planetary landing. In: Proceedings of guidance, navigation, and control conference (New Orleans, LA, U.S.A.), AIAA-97-3709, pp 1376–1381

Taylor L, Kulcinski G (1999) Helium-3 on the Moon for fusion energy: the Persian Gulf of the 21st century. Sol Syst Res 33:338

Texas Instruments (2015) Code Composer Studio (CCS) Integrated Development Environment (IDE) description. http://www.ti.com/tool/ccstudio. Accessed 8 Oct 2015

Uchiyama K, Shimada Y, Ogawa K (2005) Minimum-jerk guidance for lunar lander. Trans Jpn Soc Aeronaut Sp Sci 48(159):34

Zhang B, Tang S, Pan B (2016) Multi-constrained suboptimal powered descent guidance for lunar pinpoint soft landing. Aerospa Sci Technol 48:203

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.